

## Java POO-Encapsulation

### Exercice 1 :

Voir la source de la classe livre :

```
public class Livre {  
    // Variables  
    private String titre, auteur;  
    private int nbPages  
    // Constructeur  
    public Livre(String unAuteur, String unTitre) {  
        auteur = unAuteur;  
        titre = unTitre;  
    }  
    // Accesseur  
    public String getAuteur() {  
        return auteur;  
    }  
    // Modificateur  
    void setNbPages(int n) {  
        nbPages = nb;  
    }  
}
```

- a. Corrigez quelques petites erreurs
- b. Ajoutez une méthode main() pour Créer 2 livres, Faire afficher les auteurs de ces 2 livres.

### Exercice 2:

Accesseurs et modificateurs :

- 1) Modifiez la classe Livre :
  - a. Ajoutez un accesseur pour la variable titre et la variable nbPages.
  - b. Ajoutez un modificateur pour les variables auteur et titre.
  - c. Changez le modificateur de nbPages : il ne devra changer le nombre de pages que si on lui passe en paramètre un nombre positif, et ne rien faire sinon, en affichant un message d'erreur.
- 2) Dans la méthode main(),
  - a. indiquez le nombre de pages de chacun des 2 livres,
  - b. faites afficher ces nombres de pages,
  - c. calculez le nombre de pages total de ces 2 livres et affichez-le.

### Exercice 3:

- 1) Dans La classe Livre, ajoutez une méthode afficheToi() qui affiche une description du livre (auteur, titre et nombre de pages).
- 2) Ajoutez 2 constructeurs pour avoir 3 constructeurs dans la classe :
  - a. Un constructeur qui n'a pas de paramètre
  - b. Un qui prend en paramètre l'auteur et le titre du livre,
  - c. et l'autre qui prend en plus le nombre de pages.

Utilisez les 3 constructeurs (et éventuellement d'autres méthodes) pour créer 3 livres de 300 pages dans la méthode main() de la classe TestLivre

**Exercice 4 :**

- 1) Contrôle des variables private par les modificateurs
  - a. Ajoutez un prix aux livres (nombre qui peut avoir 2 décimales ; type Java float ou double) avec 2 méthodes getPrix et setPrix pour obtenir le prix et le modifier.
  - b. Ajoutez au moins un constructeur qui prend le prix en paramètre.
  - c. On bloque complètement les prix : un prix ne peut être saisi qu'une seule fois et ne peut être modifié ensuite (une tentative pour changer le prix ne fait qu'afficher un message d'erreur). Réécrivez la méthode setPrix (et autre chose si besoin est). Vous ajouterez une variable booléenne prixFixe (pour "prix fixé") pour savoir si le prix a déjà été fixé.
  - d. Réécrire la méthode main () et prévoir le deux cas ( prix non fixé ou bien prix fixé plusieurs fois ) afficher le résultat de l'exécution.